Set up for using *ourdata* package

Benjamin Groß

2021-12-05

Installing Dependencies

R

ourdata depends on R >= 4.0. To check your R version run `version` from the R console.

This package also depends on the following R libraries:

- `reticulate`
- `PressPurt`

If not already installed, please install these packages

```
# install PressPurt package
install.packages("PressPurt", dependencies = TRUE)
# install reticulate package
install.packages("reticulate", dependencies = TRUE)
```

Python

This package is dependent on python and uses the R package `reticulate` under the hood to use python. Thus python and the dependent packages must be installed.

There are two suggested approaches:

- Virtual environments which supports Linux/Mac OS (recommended)
- Conda (Anaconda/miniconda) which supports Linux/Mac/Windows

It is also possible to set a specific version of python with `reticulate::use_python("/path/to/python")` before you run any of the `ourdata` commands but it is not recommended. You'll also have to ensure that the dependent python packages are installed. For more information on using `reticulate` directly: see the reticulate docs.

Virtualenv

Install virtual environment

If needed, install `virtualenv` with `pip`.

Command-line

- Install, test and (optionally) create a new virtual environment from the command line

```
# install virtualenv with pip
pip install virtualenv
# test your installation
virtualenv --version
# create a virtualenv
virtualenv venv
```

Installing python dependencies

From R

The `create_virtual_env` function lets one create a virtual environment to use. If you want to make a new virtual environment, you'll need to specify which python version to use in addition to the virtual environment name.

You may check which version(s) of python you have in a fresh R session with `find_python()`. You should use the sample one you used to install `virtualenv`.

```
create_virtual_env(version = "/usr/bin/python3",
                   virtualenv = "ourdataEnv",
                   verbose = TRUE)
```

Next you'll need to install the python dependencies in your virtual environment. This will install the following into your `ourdataEnv` virtual env:

- `numpy`
- `deepl`

```
py_depend(virtualenv = "ourdataEnv")
py_install("deepl", pip = TRUE)
```

If you want to use the `ourdataEnv` environment in a new R session you can access it via:

```
set_python_virtual(virtualenv = "ourdataEnv")
```

Now that you have created the virtual environment and installed dependencies, you can use the new env in R! Run:

```
# load the PressPurt library
library(PressPurt)
# You should see "ourdataEnv" listed as a conda env
find_python()
# Set your virtualenv
set_python_virtual(virtualenv = "ourdataEnv")
```

Conda

Install Anaconda or Miniconda

Conda is a package and environment manager that is open source. The main difference between Anaconda and Miniconda is that Anaconda comes with a bundle of pre-installed packages so it takes longer to download and install. Both Miniconda and Anaconda come with python but you can specify a specific version of python later as well.

This document will show you how to install Miniconda via the command-line. For more information on installation and how to install it graphically (no command-line) see the conda docs

Linux

1. Download the Miniconda installer for Linux
2. In a terminal window run: `bash Miniconda3-latest-Linux-x86_64.sh`
3. Follow the prompts - accepting defaults should be fine.
4. Close your Terminal window and re-open a new one.
5. Test your installation by running: `conda list`
    - This should result in a list of installed packages.

MacOS

1. Download the Miniconda installer for macOS
2. In a terminal window run: `bash Miniconda3-latest-MacOSX-x86_64.sh`
3. Follow the prompts - accepting defaults should be fine.
4. Close your Terminal window and re-open a new one.
5. Test your installation by running: `conda list`
    - This should result in a list of installed packages.

Windows

1. Download the Miniconda installer for Windows
2. Run the `.exe` file by double clicking.
3. Follow the prompts - accepting defaults should be fine.
4. Test your installation by opening the Anaconda Prompt from the Start menu and running running: `conda list`
    - This should result in a list of installed packages.

Installing python dependencies

From R

First you'll need to load `PressPurt`, check which versions of python are found and if any Conda environments exists.

```
# load the PressPurt library

library(PressPurt)

# Find the versions of python, conda environments and virtualenvs available on your
machine:

find_python()
```

This shows you your default Python path, other available python paths, a list of your conda environments and virtual environments.

The `create_conda_env` function lets one create a new conda environment – you'll most likely need to specify which python version to use in addition to the conda environment name. This function also sets your conda environment to the newly created one.

You may have python installed under `/usr/bin/python3`, Anaconda2 and/or Anaconda3, so you may need to specify which python to use.

```
create_conda_env`(condaenv="ourdataEnv",

                  version="~/anaconda3/bin/python",

                  verbose = TRUE)
```

Next you'll need to install some additional python dependencies that *ourdata* needs into your conda environment. This will install the following into your `ourdataEnv` conda env:

- `numpy`
- `deepl`

```
py_depend(condaenv = "ourdataEnv")

py_install("deepl", pip = TRUE)
```

If the dependencies don't install correctly see below for instructions on how to install via the command-line.

If you want to use the `PressPurtEnv` environment in a new R session you can access it via:

```
set_python_conda(condaenv="ourdataEnv")
```

Command-line

You can also install your Python dependencies via the command line with conda or with the Anaconda Prompt (Windows).

When you activate conda it starts the “base” environment. You’ll want to create a specific environment to install all of the PressPurt requirements.

```
# Check that conda is running
conda --version
# List conda envs
conda info --envs
# Create conda environment "ourdataEnv"
conda create --name ourdataEnv
# activate ourdataEnv
source activate ourdataEnv
# check installed packages
conda list
# install python dependencies
conda install matplotlib numpy pandas scipy sympy
pip install pathos
# Check to make sure they were installed
conda list
```

Now that you have created the conda environments and installed dependencies, you can use the new environment in R. Run:

```
# load the PressPurt library
library(PressPurt)
# You should see "PressPurtEnv" listed as a conda env
find_python()
# Set your conda env
set_python_conda(condaenv="ourdataEnv")
```